

DATA COMMUNICATION LINK

Cross-Reference To Related Application

This application claims priority of Great Britain Application No. 0006291.9, which was filed on March 15, 2000.

5 Technical Field

The present invention relates to a data communication link for high speed, high bandwidth applications.

Background

In applications such as providing a data communication link between two
10 Application Specific Integrated Circuits (ASICs) in a local backplane of a computing system, very high data rates may be required, e.g. an average data rate of at least 4.8 Giga bits per second (Gbps). The data link may be 64 bits wide.

Of the various possibilities for implementing such a link, it is possible to provide an interface that transfers data from the transmitting ASIC to the receiving ASIC as a
15 single parallel word with a synchronising clock signal running at the system clock rate CK, say 78 MHz. However, for a data word of 64 bits to achieve a data transfer rate of 4.8 Gbps this would require 65 device pins, which for many applications would be either impractical or too costly to provide in the ASICs.

A synchronous interface could be used using a smaller number of pins, by
20 multiplexing a 64 bit wide data word N times onto W bits ($= 64/N$) and by providing a synchronising clock. However with a clock signal running at 78 MHz, the bandwidth would be reduced to $W * CK = BW/N$, which would give an unacceptably slow data transfer rate.

In order to achieve a bandwidth of 4.8 Gbps, the transfer rate may be multiplied N
25 times. A synchronous interface which has a resultant Transfer Clock, $N * CK$, of less than 200 MHz may be practical. Above 200 MHz, which would be necessary to achieve the desired transfer rate of 4.8 Gbps, each data bit would be valid for a maximum of 5 ns, reducing further when rise-fall times of the interconnect signals and input/output buffers are included. The task of achieving a robust design, ensuring that all W bits are aligned

such that the synchronising clock can always capture valid data bytes at the receiving ASIC, is far from trivial.

Summary of the Invention

With a view to avoiding the above noted problems, the invention provides a data
5 transmitter having: a transfer clock generator, responsive to the system clock, generating a transfer clock at a high rate relative to the system clock; and a parallel to serial register, for dividing an input word into a plurality of smaller words and transmitting them over corresponding serial sub-links in response to the transfer clock.

In an alternative embodiment, the invention further provides a receiver having: a
10 plurality of serial to parallel registers coupled to corresponding serial sub-links, for converting received serial data words from the sub-links into parallel form; and, a clock generator, responsive to the received data, for generating a low speed clock with a frequency nominally equal to the system clock.

Brief Description of the Drawing

15 A preferred embodiment of the invention will now be described with reference to the accompanying drawings wherein:-

Figure 1 is a schematic view of the transmit interface of a first ASIC of the data communication link of the invention;

Figure 2 is a schematic block diagram of the receive interface of a second ASIC
20 of the data communication link of the invention;

Figure 3 is a more detailed diagram of the control mechanism for aligning received words in each sub-link of the link of **Figures 1** and **2**; and

Figure 4 is a schematic block diagram of a Clock Data Recovery Module (CDRM) used in both the interfaces of **Figures 1** and **2**.

Description of a Preferred Embodiment

Referring to **Figures 1** and **2** of the drawing, a Link Interface between first and second ASICs 2, 4 includes an interface 6 in ASIC 2. In one embodiment:

$$DW = W * N$$

Where;

30 $DW = \text{Bit width of wide data word}$

W = Bit width of sub-data word

N = An integer value, greater than 1

The data bandwidth across the link is given as;

$$BW = DW * CK$$

5 Where;

BW = Bandwidth in Mega bits per second (Mbps)

CK = Transfer Clock in Mega Hertz (MHz)

For example, W = 8, N = 8, DW = 64. The Transfer Clock, CK, is 78 MHz giving a BW of 4992 Mbps. However the invention is not limited to these specific

10 values.

Interface 6 in ASIC 2 has a register 8 for breaking down the wide input data words, DW, into N (in this embodiment 8) smaller sub-words W (each 8 bits long). Each sub-word W is treated independently, using a Clock Data Recovery Module 10 (CDRM) macrocell. CDRM 10 has a multiplier 12 for multiplying the clock CK, W (8) times and
15 respective parallel to serial (PISO) converters 14 for operating on each of N, W bit words. Each serial word is transmitted over a respective sub-link 16.

Referring to **Figure 2**, the receive ASIC 4 has an interface 20. The serial links 16 are coupled to another CDRM macrocell 22, in which a parallel W bit word and clock is recovered for each of the N serial links.

20 **Figure 4** shows in more detail a CDRM 10, 22. The module 10, 22 has two primary functions. In transmit, it takes Low Speed Parallel Data (LDTX) on line 40 and creates High Speed Serial Data (HDTX) on line 42. In receive, it operates in reverse, taking High Speed Serial Data (HDRX) on line 44 and creating Low Speed Parallel Data (LDRX) on line 46. In addition, the receive operation also recovers a Low Speed Clock (LDCK) on line 48 from the serial data, that is phase aligned with the LDRX data. A
25 Reference System Clock (REFCK) on line 49 is applied to a Phase Locked Loop 50 which multiplies the clock rate by a factor of 8 to provide a High Speed Clock (HSCK) on line 52. HSCK is applied to a parallel to serial register 54 and to a serial to parallel register 56. HSCK is also applied to a divide by 8 unit 58 and a chain of three toggles 60.
30 The outputs of toggles 60 are detected by an edge detector device 62 which provides an

output to divider unit 58. The output of divider unit 58 comprises the Low Speed Clock (LDCK) on line 48. The operation of the circuit of **Figure 4** is as follows:

For **Transmit**, Low Speed Data (LDTX) on line 40 will be presented to the CDRM at the rate of the Reference Clock (REFCK). The Reference Clock will be multiplied in frequency eight times by Phase Locked Loop (PLL) 50 to create High Speed Clock (HSCCK) on line 52. LDTX data on line 40 will be loaded into a Parallel Serial Output (PISO) register 54 at the REFCK rate, and clocked out serially at the HSCCK rate to form HDTX data on line 42.

For **Receive**, the High Speed Clock (HSCCK) will be divided by eight at 58 to create a Low Speed Clock (LDCK) output. However, the phase of this clock must be adjusted so that its associated Low Speed Data (LDRX) is stable at the time of the active edge of LDCK. This is done by edge detection and phase adjustment circuit 60, 62 which monitors the High Speed Data (HDRX) on line 44. HDRX is also passed into a Serial Parallel Output (SIPO) register 56 to create the Low Speed Received Parallel Data (LDRX) on line 46. The output from the SIPO 56 will be enabled on the opposite edge to the active edge of its associated clock LDCK.

The number of transmit and sub-links are replicated 8 times in this example. However, there will generally only be a single PLL per CDRM macrocell.

On the receive side, the serial links are passed through CDRM macrocell 22, and a W bit word and clock will be recovered for each of the N serial links. The CDRM 22 has no knowledge of the boundary between one W bit word and the next within the serial data stream and it is therefore the first task of the Interface 20 to identify the correct bit alignment within each sub-link. Having recovered the W bit words for each sub-link, all N of the W bit words have to be aligned and synchronised to recreate the original DW width word.

The bit alignment is achieved by the transmit side sending consecutive initialisation words constructed by ASIC 2. These initialisation words (of W bits) have the property that however many times the word is shifted right or left within another word that is 2W bits wide, there is a unique position that defines the bit alignment. For

example consider an initialisation word, for $W = 8$, of “10111000”. A register 24 that is $2W$ words wide holds the previously received and

Previous & Current Word	Bit Alignment
10111000xxxxxxxx	0
x10111000xxxxxxxx	1
xx10111000xxxxxx	2
xxx10111000xxxxx	3
xxxx10111000xxxx	4
xxxxx10111000xxx	5
xxxxxx10111000xx	6
xxxxxxx10111000x	7

5 currently received words of W bits as shown in the above table. The initialisation word is sent at least twice followed by another synchronisation word (user defined) delimiter to indicate the start of transmission of true data. The position of the word is located in the register by means of a state machine (not shown) and this information is relayed to subsequent stages.

10 During transmission, each ASIC transmitting/receiving interface will respectively create/recreate a cyclic redundancy code (CRC) from the true data. The CRC words are inserted at a pre-determined interval, programmed to both transmit and receive sides. After this interval the transmitted CRC should equal the recreated CRC. If not, then
 15 the data. This provides an Integrity Check individually on each of the serial links.

Thus, as shown in **Figure 2**, subsequent to parallel conversion in CDRM 22, the parallel words are placed in a bit alignment register 24 in each sub-link for detecting bit alignment. This is effected by a state machine (not shown) locking onto the position of the initialisation word within the register, and passing the bit aligned word to the next
 20 stage. In the next stage, an Integrity Check is performed on the CRC word at 26.

The bit alignment and the Integrity Check are performed in each sub-link using the recovered clock generated for that serial link. There is no guarantee of any phase relationship between any of the N recovered clock (RCK[n])s, and each of the recovered clocks may be jittering (except that the recovered clocks will be within one clock cycle of one another). However, the average frequency of all recovered clocks and that of the Transfer Clock, CK, on the transmit side must be exactly the same, since the reference clock to both the transmit and receive ASICs will be driven from the same crystal oscillator. A mechanism is therefore required to re-align the N recovered sub-words and resynchronise the wide data word back to the Transfer Clock, CK. This is done by using a short First In First Out (FIFO) 28, 6 words long, at the end of each serial link.

The recovered sub-word plus a marker bit ($W + 1$ bits) is written to the FIFO 28 by its associated recovered clock on line 48. The marker bit indicates whether that data word was Transmitted Synchronisation or Integrity Check Word. The very first word to be written by each of the links, will be a synchronisation word (marker bit set) and the second will be the first sub-word of true data. The first write will occur at a slightly different time for each link, but by the time the second write occurs, all will have written at least once. The addressing of the FIFOs may use Johnson coding, as more clearly seen in **Figure 3**. An address generator 32 provides a Johnson scheme of addressing to the write read address 34 of the respective FIFO 28.

The initial value of the address is 011 and the address scheme changes as indicated in **Figure 3**. The most significant bit of the addresses of the sub-links are coupled by lines 36 to an OR gate 38. The output of the OR gate 38 is coupled by two metastability registers 70 to provide a trigger signal on line 72 to a state machine 74. State machine 74 provides an output on line 76 to control the reading out of the FIFO registers 28.

Thus, only a single address bit of FIFO's 28 changes per write and by ensuring that the top address bit is set on the second write, that address bit can be logically OR'd with the equivalent bit from all N links. This single bit signal, which goes high when the first word in a sub-link is received, is resynchronised via the metastability registers 70.

By this time, since it is known all FIFO registers will be written to within a clock cycle of

one another, all FIFO's will contain words, and the state machine 74 triggers the Word Aligner to read from all N FIFO's in parallel at the Transfer Clock rate, CK. This read should therefore occur when each of the FIFOs contain approximately four words. As the average frequency of the read and write clocks to the FIFO is the same each FIFO should
5 always contain approximately four words. A FIFO that is at least six deep will isolate against jitter on the recovered clocks.

The very first FIFO read will all be synchronised sub-words but the second will be the recovery of the first true wide data word. The output of the FIFOs are applied to a Word Aligner register 78 which reconstitutes the original data word 80 (**Figure 2**). Word
10 Alignment is checked at the same programmed interval used by the bit alignment, because at this time, and only at this time, all of the marker bits in each of the N FIFO's will be set.

The scheme outlined provides a robust high speed, high bandwidth local link by using a number of serial asynchronous links in parallel.

15 Thus, it will now be understood that there has been disclosed a new method and apparatus for providing a data communication link. While the invention has been particularly illustrated and described with reference to preferred embodiments thereof, it will be understood by those skilled in the art that various changes in form, details, and applications may be made therein. It is accordingly intended that the appended claims
20 shall cover all such changes in form, details and applications which do not depart from the true spirit and scope of the invention.